PicController



Manual de usuario

Raquel Sánchez Díaz

Tabla de Contenidos

Índice de figuras

1. Introducción

PicController es una interfaz gráfica desarrollada para el control de sensores y actuadores. Estos dispositivos están controlados por microcontroladores PIC, que se comunican con ellos mediante el bus I2C. A su vez los microcontroladores están comunicados entre sí mediante el bus de campo CAN.

Se utiliza el protocolo CANopen para gestionar todo lo relacionado con las comunicaciones, por lo que los PICs actuarán de esclavos y PicController será el maestro. La interfaz permite controlar y monitorizar todo el sistema. Se ha diseñado de manera que cualquier usuario pueda utilizarla, aunque no tenga conocimientos de CANopen.

Existen varias opciones configurables que se almacenarán en un directorio llamado *PicControllerConfigFiles*, el cual se crea automáticamente al ejecutar la aplicación. No se recomienda modificar directamente los parámetros de sus ficheros.

En este documento, se van a presentar brevemente las principales partes que forman la interfaz y sus funcionalidades.

2. Descripción de la ventana principal

Al arrancar la aplicación nos encontraremos lo siguiente:

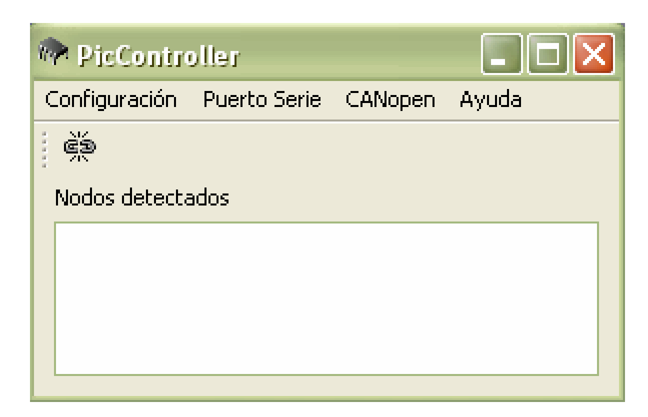


Figura 1: Aspecto de la aplicación al arrancar

En la barra de herramientas encontramos un botón que nos servirá para conectar y desconectar del puerto serie. Una vez conectados, en la lista de nodos detectados irán apareciendo los PICs que se vayan conectando al sistema, es decir, los diferentes PICs de la capa de control de hardware.

2.1. Menú Configuración

En él se puede cambiar el idioma de la interfaz. Actualmente está disponible en inglés y español. El cambio de idioma se realiza de forma instantánea y su configuración se mantiene entre las diferentes ejecuciones.

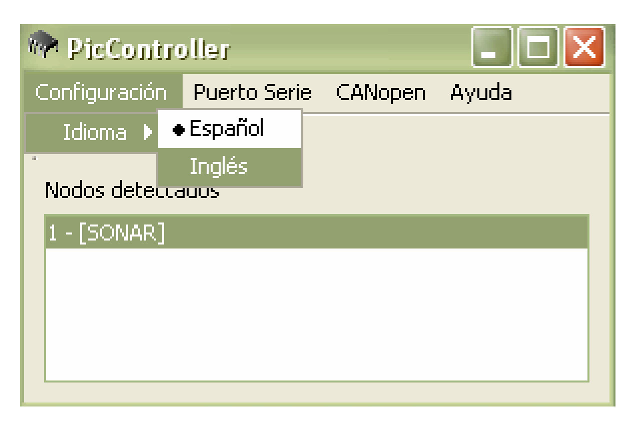


Figura 2: Configuración del idioma

2.2. Menú Puerto Serie

En este menú encontramos tres opciones:

- **Parámetros**: Permite configurar los parámetros del puerto serie. Ello se realiza mediante el cuadro de diálogo de la siguiente figura. En parámetro "Puerto" sólo aparecerán los puertos que se encuentren disponibles en el sistema en ese momento.

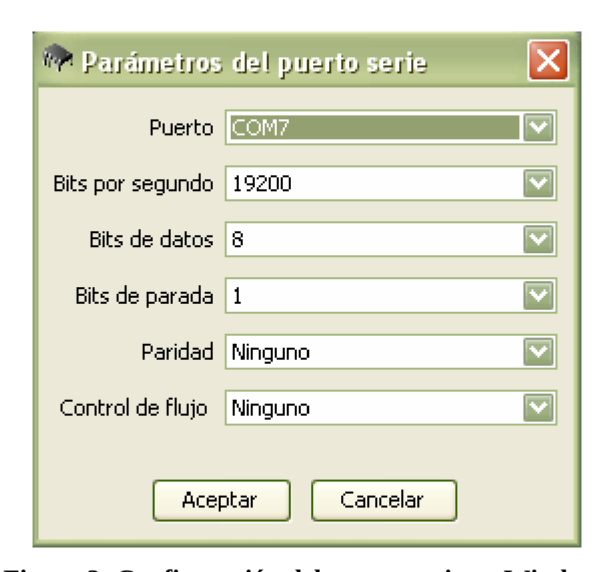


Figura 3: Configuración del puerto serie en Windows

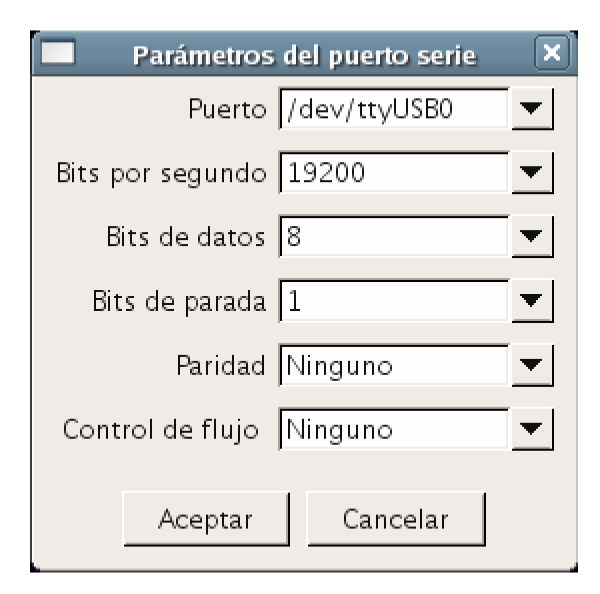


Figura 4: Configuración del puerto serie en Linux

- **Puerto serie**: Permite el acceso a la ventana que muestra toda la comunicación por el puerto serie, tal y como está ocurriendo. Desde él también se pueden enviar *bytes* de información en decimal. Deben estar separados por comas. Si estos datos se introdujeran de forma incorrecta (como muestra el ejemplo "Figura 5") se resaltarían en rojo para avisar del error.

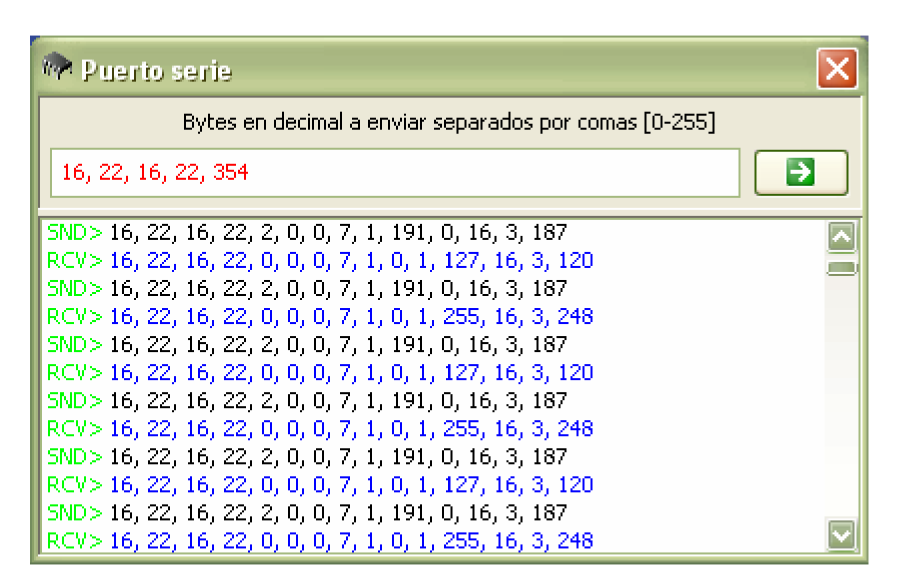


Figura 5: Comunicación por el puerto serie

- **Mensajes CAN**: Permite el acceso la ventana que muestra los mensajes CAN que se están enviando y recibiendo. Cada mensaje es desglosando en sus partes fundamentales y se muestra información sobre sus *flags* de forma textual (indica cada *flag* que esté activado). También permite enviar mensajes CAN. Si se sitúa el

puntero sobre sus campos aparecerá un *tooltip* con información sobre ellos. Un mensaje CAN tiene tres partes fundamentales:

- El identificador: Se debe introducir en hexadecimal.
- Los datos: Hasta ocho *bytes* en decimal separados por comas. Si se introdujeran erróneamente aparecerían resaltados en rojo.
- El tamaño de los datos: La interfaz lo deduce automáticamente según el se van introduciendo los datos.
- Los *flags*: Se pueden introducir de manera automática mediante casillas de verificación, o de forma manual. Es un *byte*.

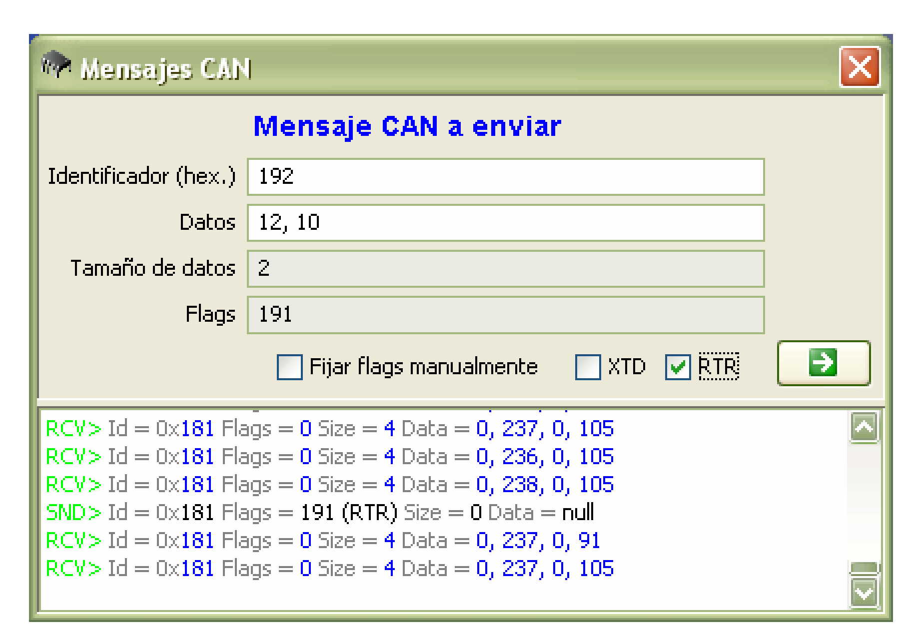


Figura 6: Mensajes CAN enviados y recibidos

2.3. Menú CANopen

Bajo este menú encontramos tres apartados:

- **Ficheros de configuración**: Esta opción sólo estará disponible cuando se hayan detectado nodos, ya que permite configurar los parámetros del diccionario de objetos de cada nodo del sistema. Esto se hace a través del cuadro de diálogo de la figura. En el cuadro combinado aparecen los nodos que están conectados actualmente en el sistema. La zona inferior va cambiando dinámicamente según se van seleccionando los nodos, de manera que muestra los objetos configurables que cada uno tiene en su diccionario. Toda esta información es almacenada dentro del directorio de configuración (*PicControllerConfigFiles*), en ficheros llamados *confFileNodeX.properties*, donde X es el número de nodo.



Figura 7: Editor de los ficheros de configuración de los nodos

- **Mensajes de error**: Permite el acceso a los mensajes de emergencia de CANopen que se han recibido. Para ello hay una ventana que se activa cada vez que llega un nuevo mensaje. En ella se muestra la información del mensaje de una manera comprensible para el usuario, aunque no tenga conocimientos sobre CANopen. Adicionalmente se puede mostrar también el mensaje CAN tal cual llega, para un usuario más avanzado.

  Tiene un menú "Edición" que permite realizar dos acciones:

    - Configurar si se debe mostrar la ventana cuando llega un nuevo mensaje de emergencia.
    - Borrar los mensajes que contiene.

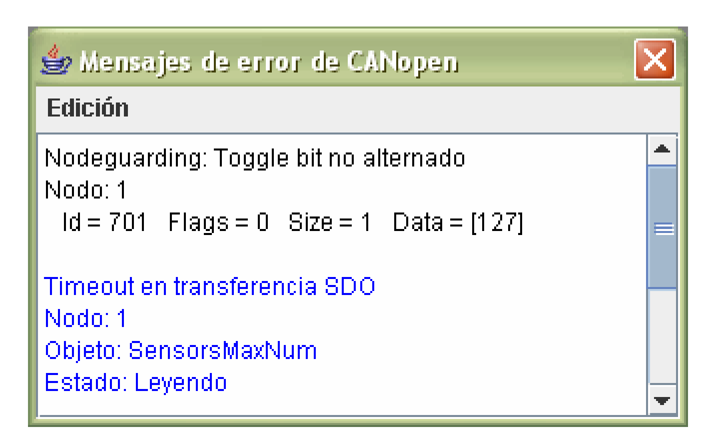


Figura 8: Mensajes de error de CANopen

- **Mensajes de alerta**: Permite acceder a otro tipo de mensajes de emergencia, las alertas. Las envían los PICs que controlan sensores cuando sus mediciones no se encuentran dentro de los límites de seguridad establecidos. Por ejemplo el nodo de los sónares las transmite al acercarse demasiado a un objeto. También en esta ventana se muestra la información de un modo fácilmente comprensible por el

usuario. Su comportamiento y estructura son los mismos que en el caso anterior, ya que ambos son mensajes de emergencia.

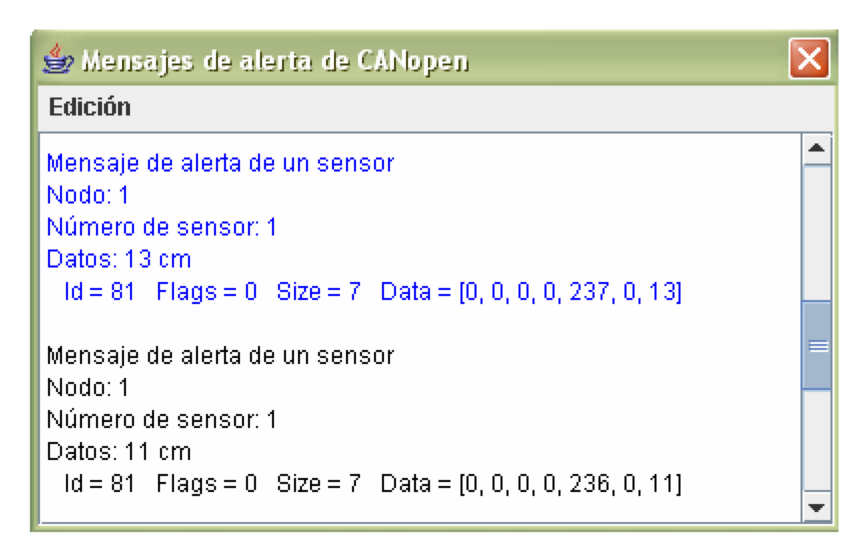


Figura 9: Mensajes de alerta de CANopen

2.4. Menú Ayuda

Contiene información sobre la herramienta.

3. Control de los PICs

Cuando un PIC se conecta al sistema aparece en la lista de nodos detectados de la ventana principal. Se muestra información sobre el número de nodo y el tipo de dispositivos que controla:

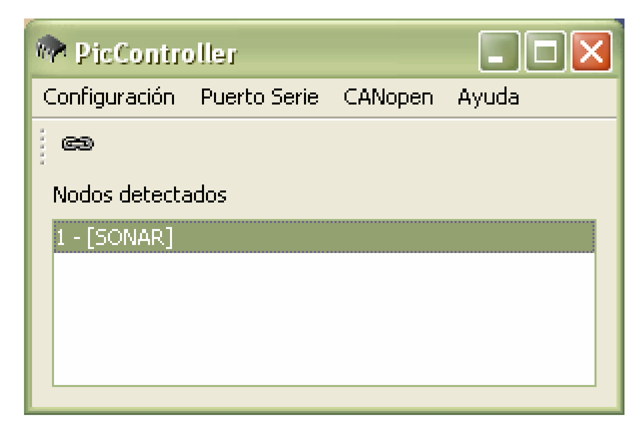


Figura 10: Ventana principal con un nodo conectado al sistema

Si se hace doble clic sobre el nodo aparecerá una ventana, específica para cada tipo de dispositivo controlado, con los objetos de su diccionario:

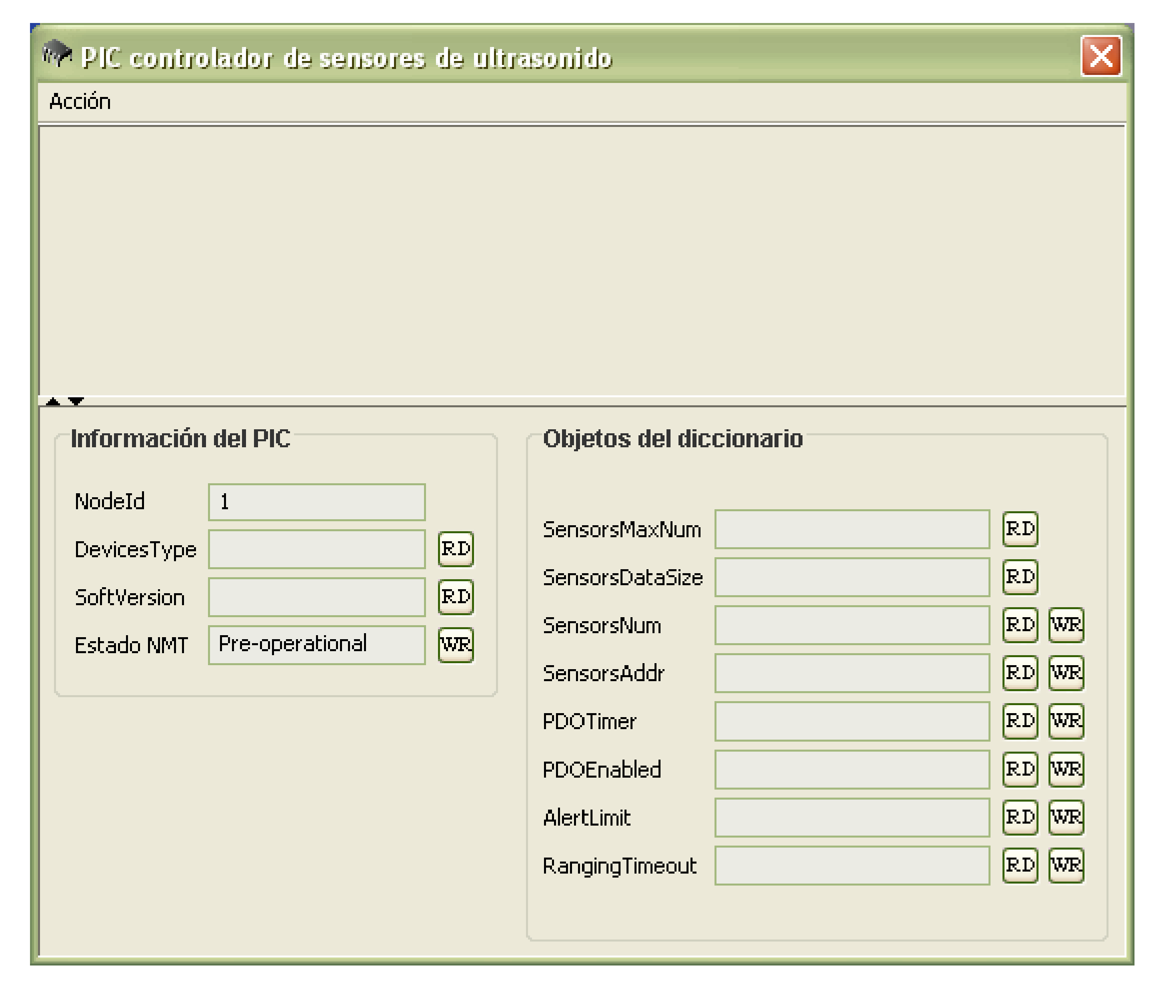


Figura 11: Controlador para sónares antes de configurar el PIC

A través de esa ventana se pueden leer y configurar los objetos del diccionario del PIC. En este caso controla sónares y como mínimo es necesario introducir la dirección de los sensores que tenga conectados para que comience a tomar medidas. Esto se puede hacer manualmente o de forma automática cargando el fichero de configuración del nodo.

Para mayor rapidez y comodidad existe un menú "**Acción**" con las siguientes opciones:

- **Cargar el fichero de configuración**: Configura el nodo. Para todos los objetos configurables del diccionario del PIC escribe los valores almacenados en el fichero de configuración propio del nodo. Esto lo realiza mediante mensajes SDO.
- **Leer todos los parámetros**: Envía todos los mensajes SDO necesarios para leer cada objeto del diccionario del PIC.
- **Enviar trama RTR**: Envía un mensaje PDO de retransmisión para que el PIC transmita los últimos valores leídos por los sensores.

También se puede realizar la lectura o escritura individual de los objetos del diccionario del PIC mediante los botones "**RD**" y "**WR**" asociados a cada parámetro. Todos los objetos que permiten su escritura directa por el usuario se modifican mediante cuadros de diálogo específicos como los que se muestran a continuación:

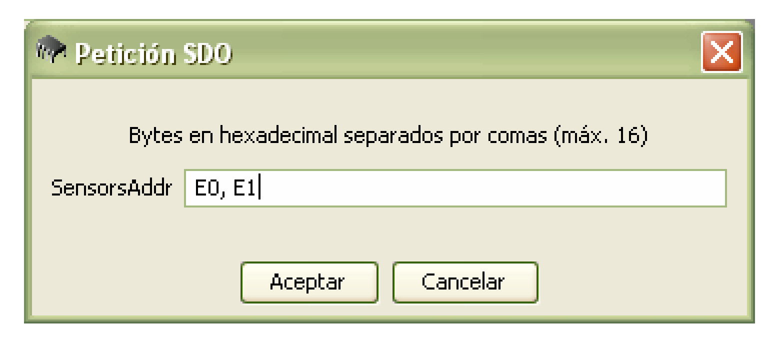


Figura 12: Configurar la dirección de los sensores



Figura 13: Configurar el objeto *PdoEnabled*

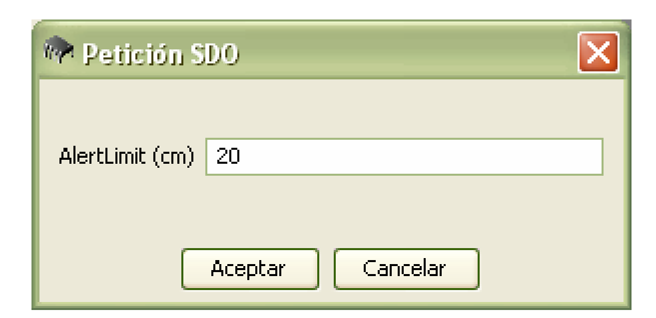


Figura 14: Configurar el objeto *AlertLimit*

Una vez configurado el PIC, éste se puede arrancar, ordenándole pasar al estado “Operational” (“Figura 16”). En ese momento comenzará a transmitir las medidas que sus sensores estén tomando. En el ejemplo “Figura 15” se puede observar un controlador para sensores SRF08 con un sólo dispositivo conectado con dirección 0xE0. El PIC ya está configurado y en estado “Operational”, por lo que se están recibiendo las medidas tomadas por el sónar.

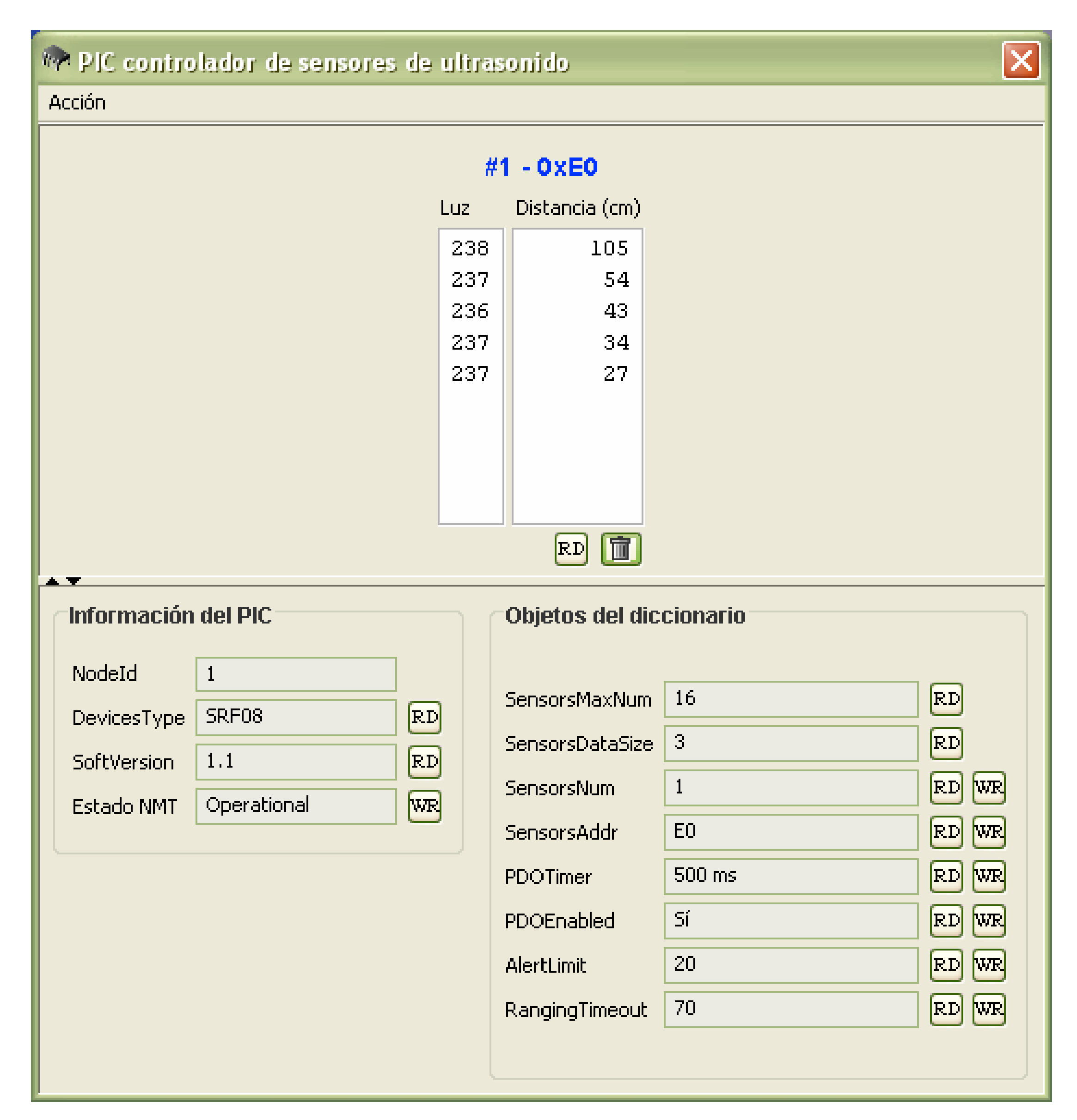


Figura 15: Ejemplo del controlador para sónares recibiendo medidas

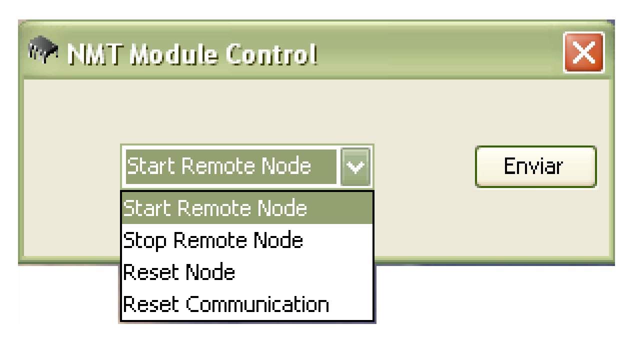
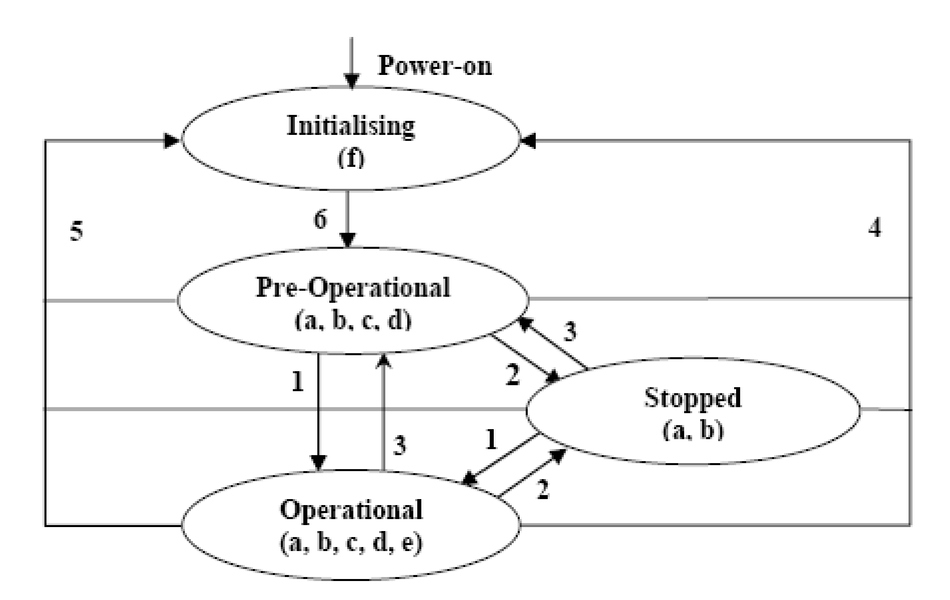


Figura 16: Cambio del estado NMT

En la figura anterior se muestra el cuadro de diálogo que permite cambiar el estado NMT del PIC. Tan sólo estarán disponibles las opciones que se puedan aplicar en cada momento. El siguiente diagrama muestra las posibles transiciones entre estados:



Las letras entre paréntesis indican qué objetos de comunicación están permitidos en cada estado:

a. NMT, b. Node Guard, c. SDO, d. Emergency, e. PDO, f. Boot-up

Transiciones entre estados (mensajes NMT):

1: Start_Remote_Node (command specifier 0x01)

2: Stop_Remote_Node (command specifier 0x02)

3: Enter_Pre-Operational_State (command specifier 0x80)

4: Reset_Node (command specifier 0x81)

5: Reset_Communication (command specifier 0x82)

6: Inicialización terminada, entra en Pre-Operational directamente al mandar el mensaje de Boot-up

Figura 17: Diagrama de transición de estados de un nodo en CANopen